

A More basics of quantum computing

This section provides more details about quantum computing and reviews the classical shadow algorithms for quantum properties estimation (QPE), along with their associated sample complexity.

Basics of quantum computation. The elementary unit of quantum computation is the qubit (or quantum bit), which is the quantum mechanical analog of a classical bit. A qubit is a two-level quantum-mechanical system described by a unit vector in the Hilbert space \mathbb{C}^2 . In Dirac notation, a qubit state is defined as $|\phi\rangle = c_0|0\rangle + c_1|1\rangle \in \mathbb{C}^2$ where $|0\rangle = [1, 0]^T$ and $|1\rangle = [0, 1]^T$ specify two unit bases and the coefficients $c_0, c_1 \in \mathbb{C}$ yield $|c_0|^2 + |c_1|^2 = 1$. Similarly, the *quantum state* of n qubits is defined as a unit vector in \mathbb{C}^{2^n} , i.e., $|\psi\rangle = \sum_{j=1}^{2^n} c_j |e_j\rangle$, where $|e_j\rangle \in \mathbb{R}^{2^n}$ is the computational basis whose j -th entry is 1 and other entries are 0, and $\sum_{j=1}^{2^n} |c_j|^2 = 1$ with $c_j \in \mathbb{C}$. Besides Dirac notation, the density matrix can be used to describe more general qubit states. For example, the density matrix of the state $|\psi\rangle$ is $\rho = |\psi\rangle\langle\psi| \in \mathbb{C}^{2^n \times 2^n}$, where $\langle\psi| = |\psi\rangle^\dagger$ refers to the complex conjugate transpose of $|\psi\rangle$. For a set of qubit states $\{p_j, |\psi_j\rangle\}_{j=1}^m$ with $p_j > 0$, $\sum_{j=1}^m p_j = 1$, and $|\psi_j\rangle \in \mathbb{C}^{2^n}$ for $j \in [m]$, its density matrix is $\rho = \sum_{j=1}^m p_j \rho_j$ with $\rho_j = |\psi_j\rangle\langle\psi_j|$ and $\text{Tr}(\rho) = 1$.

A *quantum gate* is a unitary operator that can evolve a quantum state ρ to another quantum state ρ' . Namely, an n -qubit gate $U \in \mathcal{U}(2^n)$ obeys $UU^\dagger = U^\dagger U = I_{2^n}$, where $\mathcal{U}(2^n)$ refers to the unitary group in dimension 2^n . Typical single-qubit quantum gates include the Pauli gates, which can be written as Pauli matrices:

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \quad (11)$$

The more general quantum gates are their corresponding rotation gates $R_X(\theta) = e^{-i\frac{\theta}{2}X}$, $R_Y(\theta) = e^{-i\frac{\theta}{2}Y}$, and $R_Z(\theta) = e^{-i\frac{\theta}{2}Z}$ with a tunable parameter θ , which can be written in the matrix form as

$$R_X(\theta) = \begin{bmatrix} \cos \frac{\theta}{2} & -i \sin \frac{\theta}{2} \\ -i \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{bmatrix}, \quad R_Y(\theta) = \begin{bmatrix} \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} \\ \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{bmatrix}, \quad R_Z(\theta) = \begin{bmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{bmatrix}. \quad (12)$$

They are equivalent to rotating a tunable angle θ around x , y , and z axes of the Bloch sphere, and recovering the Pauli gates X , Y , and Z when $\theta = \pi$. Moreover, a multi-qubit gate can be either an individual gate (e.g., CNOT gate) or a tensor product of multiple single-qubit gates.

The *quantum measurement* refers to the procedure of extracting classical information from the quantum state. It is mathematically specified by a Hermitian matrix H called the *observable*. Applying the observable H to the quantum state $|\psi\rangle$ yields a random variable whose expectation value is $\langle\psi|H|\psi\rangle$.

Hamiltonian and ground state. In quantum computation, a *Hamiltonian* is a Hermitian matrix that is used to characterize the evolution of a quantum system or as an observable to extract the classical information from the quantum system. Specifically, under the Schrödinger equation, a quantum gate has the mathematical form of $U = e^{-itH}$, where H is a Hermitian matrix, called the Hamiltonian of the quantum system, and t refers to the evolution time of the Hamiltonian. Typical single-qubit Hamiltonians include the Pauli matrices defined in Eqn. (11). As a result, the evolution time t refers to the tunable parameter θ in Eqn. (12). Any single-qubit Hamiltonian can be decomposed as the linear combination of Pauli matrices, i.e., $H = a_1I + a_2X + a_3Y + a_4Z$ with $a_j \in \mathbb{C}$. In the same way, a multi-qubit Hamiltonian is denoted by $H = \sum_{j=1}^{4^n} a_j P_j$, where $P_j \in \{I, X, Y, Z\}^{\otimes n}$ is the tensor product of Pauli matrices. In quantum chemistry and quantum many-body physics, the Hermitian matrix that describes the quantum system to be solved is denoted as the *problem Hamiltonian* H_C .

When taking the problem Hamiltonian as the observable, the quantum state $|\psi^*\rangle$ is said to be the *ground state* of problem Hamiltonian H if the expectation value $\langle\psi^*|H|\psi^*\rangle$ takes the minimum eigenvalue of H , which is called the *ground energy*. The ground states encode much essential information about the problem Hamiltonian, such as the critical behavior of quantum many-body systems, or the optimal solution of an optimization problem related to the problem Hamiltonian.

Numerous classical and quantum algorithms have been developed to efficiently obtain the ground states of problem Hamiltonians. These algorithms leverage various techniques, including variational

664 methods, quantum annealing, and the application of tensor networks, to approximate or directly
 665 compute the ground state. In particular, quantum algorithms such as the variational quantum
 666 eigensolver [77], quantum approximate optimization algorithm [78, 79], and the adiabatic quantum
 667 algorithm [80] show promising results for solving combinatorial optimization problems by preparing
 668 and measuring the ground state of problem Hamiltonians. The efficiency and feasibility of these
 669 methods continue to be the subject of extensive research, particularly in the context of near-term
 670 quantum devices.

671 **Classical shadow for QPE.** We review the classical shadow algorithm [13] and the sample complexity
 672 for estimating the linear and nonlinear properties of quantum states. Given a unitary operator U
 673 sampled from a unitary ensemble \mathcal{U} , and the subsequent measurement outcome \mathbf{b} under computational
 674 measurement, the classical shadow of the N -qubit quantum state ρ from m snapshots refers to

$$\hat{\rho} = \frac{1}{m} \sum_{j=1}^m \bigotimes_{i=1}^N (3U_j^{(i)\dagger} |\mathbf{b}_j^{(i)}\rangle \langle \mathbf{b}_j^{(i)}| U_j^{(i)} - I_2), \quad (13)$$

675 where random unitaries $U_j^{(i)}$ are sampled from the Pauli ensembles $\mathcal{U} = \{I_2, X, Y, Z\}^{\otimes n} \setminus I_2^{\otimes n}$. It has
 676 been shown that the sample complexity for estimating the expectation of observables O to ϵ -precision
 677 is $\mathcal{O}(4^{\text{locality}(O)} \|O\|_\infty^2 / \epsilon^2)$, with $\text{locality}(O)$ representing the number of non-identity operators acting
 678 on the qubits. For two-point correlation, the observable refers to $O = \sigma_i^x \sigma_j^x$ acting on the i -th and
 679 j -th qubits with $\text{locality}(O) = 2$, leading to the sample complexity $\mathcal{O}(16/\epsilon^2)$ for QPE.

680 The classical shadow could also be used to estimate the entanglement entropy of the quantum state ρ
 681 on the subsystem A , which is given by

$$\hat{S}_A(\rho) = -\log_2 \hat{\mathcal{P}}(\rho_A), \quad \text{with } \hat{\mathcal{P}}(\rho) = \frac{1}{m(m-1)} \sum_{j \neq j'} \text{Tr}(\hat{\rho}_A^{(j)}) \hat{\rho}_A^{(j')}, \quad (14)$$

682 where $\hat{\mathcal{P}}(\rho_A)$ refers to the purity estimation of $\text{Tr}(\rho_A^2)$, $\hat{\rho}_A^{(j)} = \bigotimes_{i \in A} (3U_j^{(i)\dagger} |\mathbf{b}_j^{(i)}\rangle \langle \mathbf{b}_j^{(i)}| U_j^{(i)} - I_2)$
 683 refers to the local classical shadow on subsystems A obtained from the j -th snapshot. It has been
 684 shown that the statistical error associated with $\hat{\mathcal{P}}(\rho_A)$ is quantified by its variance, which can be
 685 bounded as follows [81, 82],

$$\text{Var}(\hat{\mathcal{P}}(\rho_A)) \leq 4 \left(\frac{2^{|A|} \hat{\mathcal{P}}(\rho_A)}{m} \right) + 2 \left(\frac{2^{2|A|}}{m-1} \right)^2, \quad (15)$$

686 where $|A|$ denotes the number of qubits in A . This bound is known to be essentially optimal
 687 [81]. It implies that estimating the entanglement entropy requires an exponentially large number of
 688 measurements as the size of subsystem A increases.

689 B Related work

690 In this section, we present a concise review of the literature on quantum property estimation (QPE),
 691 categorizing the approaches into three primary paradigms: conventional algorithms, machine learning
 692 (ML) algorithms, and deep learning (DL) algorithms. Our discussion emphasizes that the proposed
 693 AiDE-Q framework can be seamlessly integrated with various ML and DL algorithms.

694 **Conventional algorithms for QPE.** Conventional algorithms primarily focus on estimating the
 695 quantum properties of individual quantum states. A wide range of methods have been proposed,
 696 spanning classical simulation algorithms, quantum algorithms, and quantum state learning (QSL)
 697 algorithms [83].

698 Classical simulation algorithms achieve QPE entirely on classical computers, often utilizing tensor
 699 network techniques to simulate the whole quantum state [7–11], or employing Pauli-path simulation
 700 methods to estimate properties of quantum circuit generated states [84–88]. However, these classical
 701 simulation algorithms are typically limited to specific types of quantum states, such as low-entangled
 702 or low-magic states. Quantum algorithms for QPE involve the implementation of well-designed
 703 quantum circuits to evolve quantum states and extract the desired properties. Despite their potential,
 704 these quantum algorithms often require substantial quantum resources to construct the circuits, which
 705 can hinder their practical application, especially in the early stages of quantum computing [89, 90].

Quantum state learning (QSL) algorithms for QPE operate by performing multiple measurements on a quantum state and post-processing the measurement results to estimate the quantum properties of interest [13, 20, 16, 81, 82]. Among these, the classical shadow algorithm has emerged as one of the most popular and efficient approaches, offering rigorous theoretical guarantees [13]. By using the measurement outputs from random measurement operators, this algorithm can simultaneously estimate multiple properties of quantum states, making it one of the most resource-efficient conventional algorithms for QPE.

ML algorithms for QPE. Unlike conventional algorithms that focus on QPE of individual states, machine learning algorithms address the QPE problem for classes of quantum states originating from the same quantum many-body systems with varying physical parameters. Notably, Huang et al. [91] proposed a kernel-based learning model that efficiently predicts linear properties of quantum many-body states with rigorous theoretical guarantees. This method eliminates the need for quantum devices during the prediction phase. Furthermore, Lewis et al. [36] introduced a Lasso regression model for N -qubit gapped local Hamiltonians that improves sample complexity from polynomial scaling N^c (where c is a constant) achieved in Ref. [91] to logarithmic scaling $\log(N)$.

DL algorithms for QPE. Deep learning algorithms provide enhanced capability for recognizing complex patterns in classical data collected from quantum many-body systems, enabling predictions of complex properties such as entanglement entropy with improved accuracy. Current DL algorithms can be categorized into three primary learning paradigms: supervised learning (SL), semi-supervised learning (SSL), and self-supervised learning with fine-tuning (SSL-FT).

The supervised learning paradigm explores various neural architectures for effectively constructing classical representations of the quantum states, including restricted Boltzmann machines [26, 28, 92], multi-layer perceptions [39, 40, 47], convolutional neural networks [29, 42, 46], and attention-based neural networks [35, 41, 48, 49]. The parameterized neural network for state restriction is optimized to approximate the target values of quantum properties in the training dataset under specific loss functions, such as mean square loss or cross-entropy loss.

The SSL and SSL-FT paradigms address the challenge of limited labeled data, allowing training with datasets comprising both labeled and unlabeled data. Tang et al. proposed a teacher-student model for semi-supervised training [50] and a large language model-style quantum task-agnostic pretraining and fine-tuning paradigm [51]. Both paradigms incorporate SL approaches for the labeled dataset, but differ in their sequence: SSL applies supervised learning before training with unlabeled data, whereas SSL-FT employs supervised fine-tuning after the self-supervised pretraining stage, which will be detailed in Appendix C.

C Implementation details of various learning paradigms for QPE

In this section, we present the implementation details of the three learning paradigms employed in the experiments: supervised learning (SL), semi-supervised learning (SSL), and self-supervised learning with fine-tuning (SSL-FT). All paradigms share a common neural network architecture comprising three main components: the input (encoding) layer, the hidden layer, and the output layer. To ensure a fair comparison of model performance, we adopt an identical architecture for the input and hidden layers across all learning paradigms, while the output layer is customized to suit the specific requirements of each paradigm.

C.1 Input layer and hidden layer

We follow the attention-based neural architecture proposed in Ref. [51] to design the input and hidden layers for representing quantum states.

Input layer. To capture the hidden patterns of the quantum system, the input layer incorporates three types of embeddings: token embeddings, condition embeddings, and position embeddings.

Each measurement outcome $\mathbf{o}_j \in \{0, 1\}$ under the corresponding Pauli measurement $\mathbf{M}_j \in \{X, Y, Z\}$ on the j -th qubit yields six possible combinations: $(X, 0)$, $(X, 1)$, $(Y, 0)$, $(Y, 1)$, $(Z, 0)$, and $(Z, 1)$. These pairs can be bijectively mapped to integers $\sigma \in [6]$. Consequently, the measurement outcomes (\mathbf{M}, \mathbf{o}) can be represented as a tokenized measurement string $\sigma(\mathbf{M}, \mathbf{o})$, where each element $\sigma(\mathbf{M}_i, \mathbf{o}_i) \in [6]$ resembles a token in natural language processing (NLP). The token

embedding layer applies a linear transformation to the measurement string, augmented with a start token s , producing a feature tensor $\mathbf{E}_t \in \mathbb{R}^{B_t \times (N+1) \times d}$, where B_t denotes the batch size and d is the feature dimension. A feed-forward network (FFN) with one hidden layer is then used to embed the physical condition \mathbf{p} into a vector $\mathbf{E}_c \in \mathbb{R}^{B_t \times d}$. The final input embedding is computed as the element-wise sum

$$\mathbf{E}_{\text{out}} = \mathbf{E}_t + \mathbf{E}_c + \mathbf{E}_p,$$

where \mathbf{E}_p denotes the position embeddings, as described in Ref. [93]. The resulting tensor \mathbf{E}_{out} is passed to the hidden layers for further processing.

Hidden Layer. The hidden layer consists of a multi-layer Transformer decoder, following the architecture of Ref. [93]. It takes \mathbf{E}_{out} as input and produces output $\mathbf{F} \in \mathbb{R}^{B_t \times (N+1) \times d}$, which encodes high-level representations of both the measurement strings and the conditional physical parameters. For additional architectural details, refer to Ref. [51].

C.2 Output Layer and loss function

We now separately describe the neural architectures of the output layer and the associated loss functions under the explored three learning paradigms. Specifically, the design of the output layer is tailored to accommodate the specific loss function used in each paradigm.

SL paradigms. Recall that in the SL paradigm, the training dataset for a batch is given by $\mathcal{S}_{\text{SL}} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}_{i=1}^{B_t}$, where each input $\mathbf{x}^{(i)} = (\mathbf{p}^{(i)}, \mathbf{M}^{(i)}, \mathbf{o}^{(i)})$ is derived from quantum states $\rho(\mathbf{p}^{(i)})$ with different parameters $\mathbf{p}^{(i)}$, and $\mathbf{y}^{(i)} \in \mathbb{R}^K$ represents the target label vector of dimension K . The loss function for a training batch is defined as

$$\mathcal{L}_{\text{SL}} = \frac{1}{B_t} \sum_{i=1}^{B_t} \left\| f_{\text{SL}}(\mathbf{x}^{(i)}) - \mathbf{y}^{(i)} \right\|^2. \quad (16)$$

To support this loss function, the output of the neural network for each input $\mathbf{x}^{(i)}$ must be a K -dimensional vector. Accordingly, the output layer consists of a feature aggregation module followed by a linear projection. Specifically, hidden features $\mathbf{F} \in \mathbb{R}^{B_t \times (N+1) \times d}$ are aggregated along the second axis to produce a feature representation $\mathbf{F}' \in \mathbb{R}^{B_t \times d}$ for each training example. This is followed by a linear projection into $\mathbf{F}'' \in \mathbb{R}^{B_t \times K}$, with an optional task-specific activation function. For instance, a tanh activation is used when predicting correlation functions.

SSL paradigms. In the SSL setting, the training dataset includes both labeled and unlabeled data, denoted by $\mathcal{S}_{\text{SSL}} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}_{i=1}^{n_l} \cup \{\mathbf{x}^{(i)}\}_{i=n_l+1}^n$, where n_l is the number of labeled samples. A teacher-student framework [70] is adopted, consisting of a student model f_{SSL} and a teacher model f'_{SSL} , both sharing the same architecture. The loss function for the student model in a given batch yields

$$\mathcal{L}_{\text{SSL}} = \frac{1}{B_l} \sum_{i=1}^{B_l} \left\| f_{\text{SSL}}(\mathbf{x}^{(i)}) - \mathbf{y}^{(i)} \right\|^2 + \frac{\lambda}{B_t - B_l} \sum_{i=B_l+1}^{B_t} \left\| f_{\text{SSL}}(\mathbf{x}^{(i)}) - f'_{\text{SSL}}(\mathbf{x}^{(i)}) \right\|^2, \quad (17)$$

where λ refers to the consistency weight, B_l is the number of labeled samples in the batch, and B_t is the total batch size. The proportion of labeled samples per batch is kept consistent with their proportion in the overall training dataset, i.e., $B_l/B_t = n_l/n$. The teacher model's parameters are updated as an exponential moving average of the student model's parameters, following the approach in Ref. [70].

Since the outputs of the DL models in the SSL paradigm play the same role of K -dimensional predicted labels as that in the SL paradigm, the architectures of the teacher and student networks are designed identically to the ones used in the SL setting.

SSL-FT paradigms. The SSL-FT paradigm adopts a two-stage training procedure. In the first stage, a self-supervised pretraining is performed using the unlabeled dataset $\{\mathbf{x} = (\mathbf{p}, \mathbf{M}, \mathbf{o})\}$, where the goal is to model the classical probability distribution $\mathbb{P}(\boldsymbol{\sigma}(\mathbf{M}_1, \mathbf{o}_1), \dots, \boldsymbol{\sigma}(\mathbf{M}_N, \mathbf{o}_N))$ associated with a quantum state $\rho(\mathbf{p})$ with fixed parameters \mathbf{p} . This is achieved by minimizing the average negative log-likelihood loss

$$\mathcal{L}_{\text{SF}} = \frac{1}{B_t - B_l} \sum_{i=1}^{B_t - B_l} -\log \mathbb{P} \left(\boldsymbol{\sigma}(\mathbf{M}_1^{(i)}, \mathbf{o}_1^{(i)}), \dots, \boldsymbol{\sigma}(\mathbf{M}_N^{(i)}, \mathbf{o}_N^{(i)}) \mid \mathbf{p}^{(i)} \right), \quad (18)$$

800 which corresponds to maximizing the conditional likelihood of observed measurement outcomes.

801 To produce valid probability distributions, the output layer consists of a linear transformation followed
802 by a softmax activation, ensuring normalization

$$\sum_{(\mathbf{M}_1, \mathbf{o}_1)} \cdots \sum_{(\mathbf{M}_N, \mathbf{o}_N)} \mathbb{P}(\boldsymbol{\sigma}(\mathbf{M}_1, \mathbf{o}_1), \cdots, \boldsymbol{\sigma}(\mathbf{M}_N, \mathbf{o}_N)) = 1.$$

803 In the second stage, the pretrained model is fine-tuned on the labeled dataset $\{(\mathbf{x}, \mathbf{y})\}$ using the same
804 supervised loss and output layer design as in the SL paradigm, namely

$$\mathcal{L}_{\text{FT}} = \frac{1}{B_t} \sum_{i=1}^{B_t} \left\| f_{\text{SF}}(\mathbf{x}^{(i)}) - \mathbf{y}^{(i)} \right\|^2, \quad (19)$$

805 where the initial parameters of the input and hidden layers of f_{SF} inherit the optimized parameters
806 during the pretraining stage.

807 D Integration of AiDE-Q with machine learning models

808 In this section, we briefly review the machine learning (ML) models for QPE [36, 91], and relate
809 them to the supervised learning (ML) paradigm. In this regard, the integration of ML models with
810 AiDE-Q could follow the same manner as that in the SL paradigm introduced in the main text.

811 We follow the conventions of Ref. [91] to introduce the ML model for QPE. In particular, the
812 ML model considers the training dataset $\{(\mathbf{p}^{(i)}, \hat{\rho}(\mathbf{p}^{(i)}))\}_{i=1}^n$, where $\mathbf{p}^{(i)}$ are the sampled physical
813 parameters, and $\hat{\rho}(\mathbf{p}^{(i)})$ refer to the classical shadow constructed with (\mathbf{M}, \mathbf{o}) , as defined in Eq. 13.
814 The classical ML models are trained on the size- n training data, such that when given the input
815 $\mathbf{p}^{(i)}$, the ML model can produce a classical representation $h(\mathbf{p}^{(i)})$ that approximates $\hat{\rho}(\mathbf{p}^{(i)})$. During
816 prediction, the classical ML produces $h(\mathbf{p})$ for new values of \mathbf{p} different from those in the training
817 data. In particular, the predicted output of the trained classical ML models can be written as the
818 extrapolation of the training data using a learned kernel $\kappa(\mathbf{p}, \mathbf{p}^{(i)}) \in \mathbb{R}$,

$$h(\mathbf{p}) = \frac{1}{n} \sum_{i=1}^n \kappa(\mathbf{p}, \mathbf{p}^{(i)}) \hat{\rho}(\mathbf{p}^{(i)}). \quad (20)$$

819 The ground state properties are then estimated using these predicted classical representations $h(\mathbf{p})$.
820 Specifically, $f_{\text{ML}}(\rho) = \text{tr}(O h(\mathbf{p}))$ can be predicted efficiently whenever O is a sum of few-body
821 operators. In this regard, the trained classical ML for specific properties $\hat{\mathbf{y}}^{(i)} = \text{tr}(O \hat{\rho}(\mathbf{p}^{(i)}))$ could
822 be written as

$$f_{\text{ML}}(\mathbf{p}) = \frac{1}{n} \sum_{i=1}^n \kappa(\mathbf{p}, \mathbf{p}^{(i)}) \hat{\mathbf{y}}^{(i)}. \quad (21)$$

823 Notably, the kernel function could be represented as the inner product of two feature vectors $\Phi(\mathbf{p})$
824 and $\Phi(\mathbf{p}^{(i)})$ according to Mercer's theorem [94]. In this regard, the learning model $f_{\text{ML}}(\mathbf{p})$ relates to
825 the optimal solution of the following supervised learning problem

$$f_{\text{ML}}(\mathbf{p}) = \arg \min_{f(\mathbf{p}) = \boldsymbol{\omega} \cdot \Phi(\mathbf{p})} \mathcal{L}_{\text{ML}}(\boldsymbol{\omega}) := \arg \min_{f(\mathbf{p}) = \boldsymbol{\omega} \cdot \Phi(\mathbf{p})} \frac{1}{n} \sum_{i=1}^n \left(f(\mathbf{p}^{(i)}) - \hat{\mathbf{y}}^{(i)} \right)^2, \quad (22)$$

826 where $\boldsymbol{\omega}$ refers to the optimized parameters. For neural tangent kernel $\kappa(\mathbf{p}, \mathbf{p}^{(i)})$ [37], the feature
827 vector $\Phi(\mathbf{p})$ corresponds to the output of a deep neural network with large hidden layers [95].

828 To summarize, the ML models used in QPE naturally fit within the supervised learning paradigm
829 described in Appendix C. Consequently, integrating these ML models with AiDE-Q can proceed in
830 the same manner as outlined for the SL paradigm in the main text.

831 E Experimental setting and more experimental results

832 E.1 Experimental setting

833 **Hardware platform.** All the generation of training datasets are implemented by PastaQ [96] and
834 ITensors [97] in the Julia language and run on classical device with Intel(R) Xeon(R) Gold 6226R

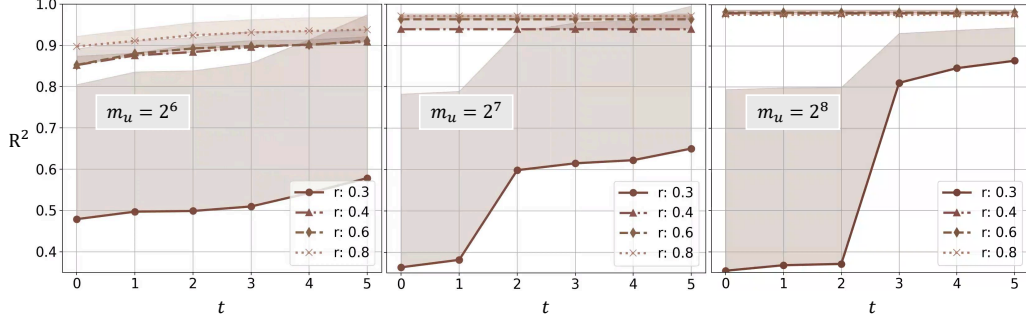


Figure 5: R^2 in entanglement entropy prediction for the 50-qubit Heisenberg XXZ model without using the physical parameters for constructing the training dataset. The panels from left to right correspond to the prediction performance for the number of measurements $m_u \in \{2^6, 2^7, 2^8\}$.

CPU @ 2.90GHz and 256 GB memory. All deep learning models in various learning paradigms are implemented by Pytorch [98] and are trained on a single NVIDIA GeForce RTX 3090 with 24G graphics memory.

Hyperparameter setting. The deep learning (DL) models use the attention-based neural architectures described in Appendix C. The embedding dimension is set to $d = 128$, the number of attention heads is set to 4, and the number of hidden layers is set to $L = 2$. We optimize the DL model using the ADAM optimizer with a learning rate of $l = 10^{-3}$ and a batch size of $B_t = 64$. The maximum number of training epochs is set to 300. To mitigate overfitting, we employ an early stopping strategy. During the initial training stage, early stopping is applied after 100 epochs. In the subsequent training stage, with the updated high-quality dataset, early stopping is initiated after 30 epochs. Each configuration is run 5 times to report the average prediction performance.

E.2 More experimental results for the Heisenberg XXZ model

In this section, we examine the prediction performance of the deep learning (DL) model trained on a dataset that does not include physical parameters as input. Specifically, instead of using the training dataset with inputs $\mathbf{x}(\mathbf{p}^{(i)}) = (\mathbf{p}^{(i)}, \mathbf{o}^{(i)}, \mathbf{M}^{(i)})$ as defined in Eq.(2), we construct the dataset with inputs $\mathbf{x}^{(i)} = (\mathbf{o}^{(i)}, \mathbf{M}^{(i)})$, where the physical parameters $\mathbf{p}^{(i)}$ are assumed to be unknown during the training process. The experimental results for predicting the entanglement entropy of the 50-qubit Heisenberg XXZ model are presented in Fig. 5. It is observed that the prediction performance can be consistently improved by AiDE-Q when the dataset contains a small ratio of high-quality data. For instance, with a large number of measurements ($m_u = 2^8$) or low-quality data and a small ratio of high-quality data ($r = 0.3$), the coefficient of determination R^2 improves from approximately 0.36 to 0.88. These experimental results demonstrate the effectiveness of AiDE-Q for handling various types of training data.

E.3 More experimental results for the cluster-Ising model

In this section, we present additional experimental results for the 9-qubit cluster Ising model, following the same format as the Heisenberg XXZ model results discussed in the main text. The hyperparameter settings used to construct the training dataset are consistent with those described in the main text.

Supervised learning models integrated with AiDE-Q outperform all reference models. Table 3 presents the coefficient of determination R^2 for predicting entanglement entropy and two-point correlations in the 9-qubit cluster Ising model. The results show similar trends to those observed for the Heisenberg XXZ model, where the baseline SL model integrated with AiDE-Q achieves a significant performance improvement compared to the standalone SL model, outperforming all reference learning models across different properties and various ratio settings. Additionally, when the ratio of high-quality data is large $r = 0.8$, the baseline SL model even surpasses the advanced SSL4Q and LLM4QPE models in predicting entanglement entropy. This suggests that directly training DL models with low-quality data can harm prediction performance, even when using advanced

Table 3: R^2 in predicting the entanglement entropy S_A , two-point correlations C_{1j}^x and C_{1j}^z of 9-qubit cluster Ising model, where $A = [j]$ and $j \in [N - 1]$. The number of measurements for low-quality data is set as $m_u = 2^6$. The best results are emphasized in blue while the second-best results are distinguished in orange.

Method	S_A			C_{1j}^x			C_{1j}^z		
	$r = 0.4$	$r = 0.6$	$r = 0.8$	$r = 0.4$	$r = 0.6$	$r = 0.8$	$r = 0.4$	$r = 0.6$	$r = 0.8$
SL	0.226	0.356	0.51	0.902	0.928	0.938	0.903	0.925	0.958
SSL4Q	0.422	0.448	0.489	0.948	0.932	0.952	0.958	0.964	0.952
LLM4QPE	0.29	0.322	0.436	0.902	0.922	0.949	0.909	0.948	0.954
NTK	-	-	-	0.962	0.966	0.97	0.291	0.31	0.341
CS	-	-	-	0.829	0.829	0.829	0.	0.	0.
SL w. DE	0.563	0.791	0.873	0.956	0.965	0.976	0.965	0.98	0.984

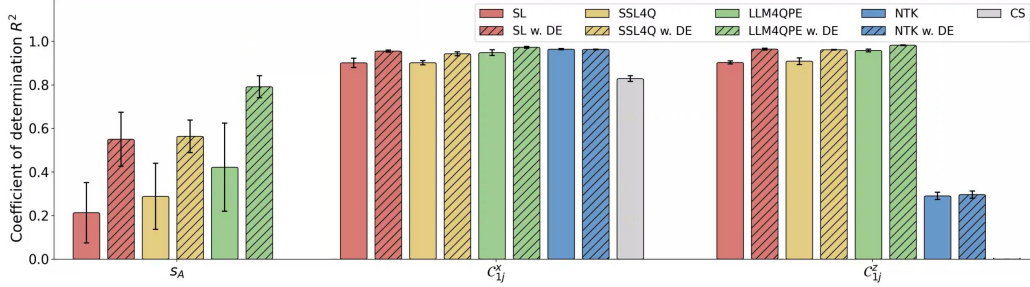


Figure 6: R^2 of reference models with and without integrating AiDE-Q in predicting entanglement entropy S_A , two-point correlations C_{1j}^x and C_{1j}^z of 9-qubit cluster Ising model, where $A = [j]$ and $j \in [N - 1]$. The initial ratio of high-quality data and the number of measurements for low-quality data is set as $r = 0.4$ and $m_u = 2^6$.

learning models. In contrast, the AiDE-Q-enhanced SL model, trained with the identified high-quality synthetic labeled data, significantly improves the best R^2 value across the reference models from 0.51 to 0.873. These results emphasize the importance of constructing high-quality synthetic labeled data when training DL models for quantum property estimation.

AiDE-Q enhances the prediction performance of various DL-based models. Fig. 6 presents numerical results of predicting quantum properties when integrating AiDE-Q into various learning models. The ratio of high-quality data is set to $r = 0.4$, and the number of measurements for low-quality data is $m_u = 2^6$. The results show that all learning models exhibit an improvement in R^2 after incorporating AiDE-Q into their baseline models, particularly in predicting the complex non-linear properties of entanglement entropy.

E.4 Experimental results for the chemical molecular model

We conduct numerical simulations for predicting the properties of chemical molecular models, focusing on the H_4 molecule with varied inter-atomic length.

Dataset Generation. The Hamiltonian of a molecular system under Jordan-Wigner transformation is given by

$$H(\mathbf{p}) = \sum_{i=1}^N \sum_{\alpha \in \{x,y,z\}} c_i^\alpha(\mathbf{p}) \sigma_i^\alpha + \sum_{i,j=1}^N \sum_{\alpha,\beta \in \{x,y,z\}} c_{i,j}^{(\alpha,\beta)}(\mathbf{p}) \sigma_i^\alpha \sigma_j^\beta, \quad (23)$$

where σ_i^z (σ_i^x) is the Pauli-Z (Pauli-X) operator acting on the i -th qubit, $c_i^\alpha(\mathbf{p})$, $c_{i,j}^{(\alpha,\beta)}(\mathbf{p})$ refer to the Pauli coefficients dependent on the inter-atomic length \mathbf{p} . This Hamiltonian could be constructed with the PennyLane package [99]. For the H_4 molecule, this system is characterized by an 8-qubit Hamiltonian $H(\mathbf{p})$. Here, the inter-atomic length \mathbf{p} is sampled from the range 0.7\AA to 1.3\AA . The number of sampled inter-atomic lengths is set as $n = 1280$. Other hyperparameters used to build the hybrid dataset in Eq. (5) are set to be the same as those used in quantum many-body systems.

Experimental results. Table 4 shows the coefficient of determination R^2 of entanglement entropy prediction for the H_4 molecule model. The results demonstrate that integrating AiDE-Q into various

Table 4: R^2 in predicting the entanglement entropy S_A of H_4 molecular systems with $A = [j]$ and $j \in [N - 1]$. The best results are emphasized in blue while the second-best results are distinguished in orange.

Method	$m_u = 2^6$			$m_u = 2^7$			$m_u = 2^8$		
	$r = 0.4$	$r = 0.6$	$r = 0.8$	$r = 0.4$	$r = 0.6$	$r = 0.8$	$r = 0.4$	$r = 0.6$	$r = 0.8$
SL	0.085	0.178	0.178	0.196	0.254	0.275	0.127	0.167	0.225
SSL4Q	0.251	0.109	0.404	0.203	0.224	0.408	0.082	0.165	0.305
LLM4QPE	0.033	0.204	0.238	0.098	0.144	0.205	0.077	0.186	0.243
SL w. DE	0.405	0.619	0.679	0.478	0.638	0.734	0.5	0.632	0.751
SSL4Q w. DE	0.495	0.508	0.709	0.507	0.546	0.704	0.543	0.57	0.694
LLM4QPE w. DE	0.293	0.58	0.618	0.473	0.572	0.638	0.502	0.584	0.64

learning-based models significantly enhances their predictive performance across different ratios of high-quality data $r \in \{0.4, 0.6, 0.8\}$ and measurement counts $m_u \in \{2^6, 2^7, 2^8\}$. In particular, the mean R^2 values for the SL, SSL4Q, and LLM4QPE models are improved from 0.187, 0.239, 0.158 to 0.604, 0.586, 0.544, respectively, after integrating AiDE-Q. These results confirm the effectiveness of AiDE-Q in enhancing the prediction ability of DL models for a variety of quantum systems.

References

- [1] Henrik Bruus and Karsten Flensberg. *Many-body quantum theory in condensed matter physics: an introduction*. Oxford university press, 2004.
- [2] Bela Bauer, Sergey Bravyi, Mario Motta, and Garnet Kin-Lic Chan. Quantum algorithms for quantum chemistry and quantum materials science. *Chemical reviews*, 120(22):12685–12717, 2020.
- [3] Philippe Andre Martin and François Rothen. *Many-body problems and quantum field theory: an introduction*. Springer Science & Business Media, 2013.
- [4] Thomas Frauenheim, Gotthard Seifert, Marcus Elstner, Thomas Niehaus, Christof Köhler, Marc Amkreutz, Michael Sternberg, Zoltán Hajnal, Aldo Di Carlo, and Sándor Suhai. Atomistic simulations of complex materials: ground-state and excited-state properties. *Journal of Physics: Condensed Matter*, 14(11):3015, 2002.
- [5] Ingrid Rotter and JP Bird. A review of progress in the physics of open quantum systems: theory and experiment. *Reports on Progress in Physics*, 78(11):114001, 2015.
- [6] Thomas D Barrett, Aleksei Malyshev, and AI Lvovsky. Autoregressive neural-network wave-functions for ab initio quantum chemistry. *Nature Machine Intelligence*, 4(4):351–358, 2022.
- [7] Steven R White. Density matrix formulation for quantum renormalization groups. *Physical review letters*, 69(19):2863, 1992.
- [8] Walter Kohn. Nobel lecture: Electronic structure of matter—wave functions and density functionals. *Reviews of modern physics*, 71(5):1253, 1999.
- [9] Román Orús. Tensor networks for complex quantum systems. *Nature Reviews Physics*, 1(9):538–550, 2019.
- [10] Feng Pan and Pan Zhang. Simulation of quantum circuits using the big-batch tensor network method. *Physical Review Letters*, 128(3):030501, 2022.
- [11] Eric R Anschuetz, Andreas Bauer, Bobak T Kiani, and Seth Lloyd. Efficient classical algorithms for simulating symmetric quantum systems. *Quantum*, 7:1189, 2023.
- [12] Federico Becca and Sandro Sorella. *Quantum Monte Carlo approaches for correlated systems*. Cambridge University Press, 2017.
- [13] Hsin-Yuan Huang, Richard Kueng, and John Preskill. Predicting many properties of a quantum system from very few measurements. *Nature Physics*, 16(10):1050–1057, 2020.
- [14] GI Struchalin, Ya A Zagorovskii, EV Kovlakov, SS Straupe, and SP Kulik. Experimental estimation of quantum state properties from classical shadows. *PRX Quantum*, 2(1):010307, 2021.
- [15] Andreas Elben, Steven T Flammia, Hsin-Yuan Huang, Richard Kueng, John Preskill, Benoît Vermersch, and Peter Zoller. The randomized measurement toolbox. *Nature Reviews Physics*, 5(1):9–24, 2023.
- [16] Anurag Anshu and Srinivasan Arunachalam. A survey on the complexity of learning quantum states. *Nature Reviews Physics*, 6(1):59–69, 2024.
- [17] Haimeng Zhao, Laura Lewis, Ishaan Kannan, Yihui Quek, Hsin-Yuan Huang, and Matthias C Caro. Learning quantum states and unitaries of bounded gate complexity. *PRX Quantum*, 5(4):040306, 2024.
- [18] Xiaosi Xu, Simon Benjamin, Jinzhao Sun, Xiao Yuan, and Pan Zhang. A herculean task: Classical simulation of quantum computers. *arXiv preprint arXiv:2302.08880*, 2023.
- [19] Ashwin Nayak and Angus Lowe. Lower bounds for learning quantum states with single-copy measurements. *ACM Transactions on Computation Theory*, 2025.

- [20] Tiff Brydges, Andreas Elben, Petar Jurcevic, Benoît Vermersch, Christine Maier, Ben P Lanyon, Peter Zoller, Rainer Blatt, and Christian F Roos. Probing rényi entanglement entropy via randomized measurements. *Science*, 364(6437):260–263, 2019.
- [21] Andreas Elben, Benoit Vermersch, Christian F Roos, and Peter Zoller. Statistical correlations between locally randomized measurements: A toolbox for probing entanglement in many-body quantum states. *Physical Review A*, 99(5):052323, 2019.
- [22] Xun Gao and Lu-Ming Duan. Efficient representation of quantum many-body states with deep neural networks. *Nature communications*, 8(1):662, 2017.
- [23] Giuseppe Carleo and Matthias Troyer. Solving the quantum many-body problem with artificial neural networks. *Science*, 355(6325):602–606, 2017.
- [24] Juan Carrasquilla and Roger G Melko. Machine learning phases of matter. *Nature Physics*, 13(5):431–434, 2017.
- [25] Patrick Huembeli, Alexandre Dauphin, and Peter Wittek. Identifying quantum phase transitions with adversarial neural networks. *Physical Review B*, 97(13):134109, 2018.
- [26] Giacomo Torlai, Guglielmo Mazzola, Juan Carrasquilla, Matthias Troyer, Roger Melko, and Giuseppe Carleo. Neural-network quantum state tomography. *Nature physics*, 14(5):447–450, 2018.
- [27] Andrea Rocchetto, Edward Grant, Sergii Strelchuk, Giuseppe Carleo, and Simone Severini. Learning hard quantum distributions with variational autoencoders. *npj Quantum Information*, 4(1):28, 2018.
- [28] Juan Carrasquilla, Giacomo Torlai, Roger G Melko, and Leandro Aolita. Reconstructing quantum states with generative models. *Nature Machine Intelligence*, 1(3):155–161, 2019.
- [29] Benno S Rem, Niklas Käming, Matthias Tarnowski, Luca Asteria, Nick Fläschner, Christoph Becker, Klaus Sengstock, and Christof Weitenberg. Identifying quantum phase transitions using artificial neural networks on experimental data. *Nature Physics*, 15(9):917–920, 2019.
- [30] Giacomo Torlai, Brian Timar, Evert PL Van Nieuwenburg, Harry Levine, Ahmed Omran, Alexander Keesling, Hannes Bernien, Markus Greiner, Vladan Vuletić, Mikhail D Lukin, et al. Integrating neural networks with a quantum simulator for state reconstruction. *Physical review letters*, 123(23):230504, 2019.
- [31] Hsin-Yuan Huang, Richard Kueng, Giacomo Torlai, Victor V Albert, and John Preskill. Provably efficient machine learning for quantum many-body problems. *Science*, 377(6613):eabk3333, 2022.
- [32] Yan Zhu, Ya-Dong Wu, Ge Bai, Dong-Sheng Wang, Yuexuan Wang, and Giulio Chiribella. Flexible learning of quantum states with generative query neural networks. *Nature communications*, 13(1):6222, 2022.
- [33] Cole Miles, Rhine Samajdar, Sepehr Ebadi, Tout T Wang, Hannes Pichler, Subir Sachdev, Mikhail D Lukin, Markus Greiner, Kilian Q Weinberger, and Eun-Ah Kim. Machine learning discovery of new phases in programmable quantum simulator snapshots. *Physical Review Research*, 5(1):013026, 2023.
- [34] Haoxiang Wang, Maurice Weber, Josh Izaac, and Cedric Yen-Yu Lin. Predicting properties of quantum systems with conditional generative models. *arXiv preprint arXiv:2211.16943*, 2022.
- [35] Yuan-Hang Zhang and Massimiliano Di Ventra. Transformer quantum state: A multipurpose model for quantum many-body problems. *Physical Review B*, 107(7):075147, 2023.
- [36] Laura Lewis, Hsin-Yuan Huang, Viet T Tran, Sebastian Lehner, Richard Kueng, and John Preskill. Improved machine learning algorithm for predicting ground state properties. *nature communications*, 15(1):895, 2024.

- [37] Marc Wanner, Laura Lewis, Chiranjib Bhattacharyya, Devdatt Dubhashi, and Alexandru Gheorghiu. Predicting ground state properties: Constant sample complexity and deep learning algorithms. *arXiv preprint arXiv:2405.18489*, 2024.
- [38] Andi Gu, Lukasz Cincio, and Patrick J Coles. Practical hamiltonian learning with unitary dynamics and gibbs states. *Nature Communications*, 15(1):312, 2024.
- [39] Jun Gao, Lu-Feng Qiao, Zhi-Qiang Jiao, Yue-Chi Ma, Cheng-Qiu Hu, Ruo-Jing Ren, Ai-Lin Yang, Hao Tang, Man-Hong Yung, and Xian-Min Jin. Experimental machine learning of quantum states. *Physical review letters*, 120(24):240501, 2018.
- [40] Xiaoqian Zhang, Maolin Luo, Zhaodi Wen, Qin Feng, Shengshi Pang, Weiqi Luo, and Xiaoqi Zhou. Direct fidelity estimation of quantum states using machine learning. *Physical Review Letters*, 127(13):130503, 2021.
- [41] Peter Cha, Paul Ginsparg, Felix Wu, Juan Carrasquilla, Peter L McMahon, and Eun-Ah Kim. Attention-based quantum tomography. *Machine Learning: Science and Technology*, 3(1):01LT01, 2021.
- [42] Tobias Schmale, Moritz Reh, and Martin Gärttner. Efficient quantum state tomography with convolutional neural networks. *npj Quantum Information*, 8(1):115, 2022.
- [43] Tailong Xiao, Jingzheng Huang, Hongjing Li, Jianping Fan, and Guihua Zeng. Intelligent certification for quantum simulators via machine learning. *npj Quantum Information*, 8(1):138, 2022.
- [44] Yulei Huang, Liangyu Che, Chao Wei, Feng Xu, Xinfang Nie, Jun Li, Dawei Lu, and Tao Xin. Measuring quantum entanglement from local information by machine learning. *arXiv preprint arXiv:2209.08501*, 2022.
- [45] Korbinian Kottmann, Patrick Huembeli, Maciej Lewenstein, and Antonio Acín. Unsupervised phase discovery with deep anomaly detection. *Physical Review Letters*, 125(17):170603, 2020.
- [46] Ya-Dong Wu, Yan Zhu, Ge Bai, Yuexuan Wang, and Giulio Chiribella. Quantum similarity testing with convolutional neural networks. *Physical Review Letters*, 130(21):210601, 2023.
- [47] Ya-Dong Wu, Yan Zhu, Yuexuan Wang, and Giulio Chiribella. Learning quantum properties from short-range correlations using multi-task networks. *Nature Communications*, 15(1):8796, 2024.
- [48] Yuxuan Du, Yibo Yang, Tongliang Liu, Zhouchen Lin, Bernard Ghanem, and Dacheng Tao. Shadownet for data-centric quantum system learning. *arXiv preprint arXiv:2308.11290*, 2023.
- [49] Yang Qian, Yuxuan Du, Zhenliang He, Min-Hsiu Hsieh, and Dacheng Tao. Multimodal deep representation learning for quantum cross-platform verification. *Physical Review Letters*, 133(13):130601, 2024.
- [50] Yehui Tang, Nianzu Yang, Mabiao Long, and Junchi Yan. Ssl4q: Semi-supervised learning of quantum data with application to quantum state classification. In *Forty-first International Conference on Machine Learning*, 2024.
- [51] Yehui Tang, Hao Xiong, Nianzu Yang, Tailong Xiao, and Junchi Yan. Towards llm4qpe: Unsupervised pretraining of quantum property estimation and a benchmark. In *The Twelfth International Conference on Learning Representations*, 2024.
- [52] Yehui Tang, Mabiao Long, and Junchi Yan. Quadim: A conditional diffusion model for quantum state property estimation. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [53] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.
- [54] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE transactions on neural networks and learning systems*, 34(11):8135–8153, 2022.

- [55] Daniel A Roberts, Sho Yaida, and Boris Hanin. *The principles of deep learning theory*, volume 46. Cambridge University Press Cambridge, MA, USA, 2022.
- [56] Vishal Goar and Nagendra Singh Yadav. Foundations of machine learning. In *Intelligent Optimization Techniques for Business Analytics*, pages 25–48. IGI Global, 2024.
- [57] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.
- [58] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.
- [59] Mingfu Liang, Jong-Chyi Su, Samuel Schuster, Sparsh Garg, Shiyu Zhao, Ying Wu, and Manmohan Chandraker. Aide: An automatic data engine for object detection in autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14695–14706, 2024.
- [60] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10371–10381, 2024.
- [61] Youcai Zhang, Xinyu Huang, Jinyu Ma, Zhaoyang Li, Zhaochuan Luo, Yanchun Xie, Yuzhuo Qin, Tong Luo, Yaqian Li, Shilong Liu, et al. Recognize anything: A strong image tagging model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1724–1732, 2024.
- [62] Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024.
- [63] Daochen Zha, Zaid Pervaiz Bhat, Kwei-Herng Lai, Fan Yang, Zhimeng Jiang, Shaochen Zhong, and Xia Hu. Data-centric artificial intelligence: A survey. *ACM Computing Surveys*, 57(5): 1–42, 2025.
- [64] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge University Press, 2010.
- [65] Matthew Rispoli, Alexander Lukin, Robert Schittko, Sooshin Kim, M Eric Tai, Julian Léonard, and Markus Greiner. Quantum critical behaviour at the many-body localization transition. *Nature*, 573(7774):385–389, 2019.
- [66] Luigi Amico, Rosario Fazio, Andreas Osterloh, and Vlatko Vedral. Entanglement in many-body systems. *Reviews of modern physics*, 80(2):517, 2008.
- [67] Mark M Wilde. *Quantum information theory*. Cambridge University Press, 2013.
- [68] John Watrous. *The theory of quantum information*. Cambridge university press, 2018.
- [69] Dominik Koutný, Laia Ginés, Magdalena Moczala-Dusanowska, Sven Höfling, Christian Schneider, Ana Predojević, and Miroslav Ježek. Deep learning of quantum entanglement from incomplete measurements. *Science Advances*, 9(29):eadd7131, 2023.
- [70] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017.
- [71] Andreas Elben, Jinlong Yu, Guanyu Zhu, Mohammad Hafezi, Frank Pollmann, Peter Zoller, and Benoît Vermersch. Many-body topological invariants from randomized measurements in synthetic quantum matter. *Science advances*, 6(15):eaaz3666, 2020.

- [72] Marko Žnidarič, Tomaž Prosen, and Peter Prelovšek. Many-body localization in the heisenberg xxz magnet in a random field. *Physical Review B—Condensed Matter and Materials Physics*, 77(6):064426, 2008.
- [73] Pietro Smacchia, Luigi Amico, Paolo Facchi, Rosario Fazio, Giuseppe Florio, Saverio Pascazio, and Vlatko Vedral. Statistical mechanics of the cluster ising model. *Physical Review A—Atomic, Molecular, and Optical Physics*, 84(2):022304, 2011.
- [74] Naeimeh Mohseni, Peter L McMahon, and Tim Byrnes. Ising machines as hardware solvers of combinatorial optimization problems. *Nature Reviews Physics*, 4(6):363–379, 2022.
- [75] Fred Glover, Gary Kochenberger, and Yu Du. A tutorial on formulating and using qubo models. *arXiv preprint arXiv:1811.11538*, 2018.
- [76] Kotaro Tanahashi, Shinichi Takayanagi, Tomomitsu Motohashi, and Shu Tanaka. Application of ising machines and a software development for ising machines. *Journal of the Physical Society of Japan*, 88(6):061010, 2019.
- [77] Jules Tilly, Hongxiang Chen, Shuxiang Cao, Dario Picozzi, Kanav Setia, Ying Li, Edward Grant, Leonard Wossnig, Ivan Rungger, George H Booth, et al. The variational quantum eigensolver: a review of methods and best practices. *Physics Reports*, 986:1–128, 2022.
- [78] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.
- [79] Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D Lukin. Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices. *Physical Review X*, 10(2):021067, 2020.
- [80] Tameem Albash and Daniel A Lidar. Adiabatic quantum computation. *Reviews of Modern Physics*, 90(1):015002, 2018.
- [81] Andreas Elben, Richard Kueng, Hsin-Yuan Huang, Rick van Bijnen, Christian Kokail, Marcello Dalmonte, Pasquale Calabrese, Barbara Kraus, John Preskill, Peter Zoller, et al. Mixed-state entanglement from local randomized measurements. *Physical Review Letters*, 125(20):200501, 2020.
- [82] Benoît Vermersch, Marko Ljubotina, J Ignacio Cirac, Peter Zoller, Maksym Serbyn, and Lorenzo Piroli. Many-body entropies and entanglement from polynomially many local measurements. *Physical Review X*, 14(3):031035, 2024.
- [83] Yinfei Li, Sanjib Ghosh, Jiangwei Shang, Qihua Xiong, and Xiangdong Zhang. Estimating many properties of a quantum state via quantum reservoir processing. *Physical Review Research*, 6(1):013211, 2024.
- [84] Enrico Fontana, Manuel S Rudolph, Ross Duncan, Ivan Rungger, and Cristina Cîrstoiu. Classical simulations of noisy variational quantum circuits. *arXiv preprint arXiv:2306.05400*, 2023.
- [85] Manuel S Rudolph, Enrico Fontana, Zoë Holmes, and Lukasz Cincio. Classical surrogate simulation of quantum systems with lowesa. *arXiv preprint arXiv:2308.09109*, 2023.
- [86] Armando Angrisani, Alexander Schmidhuber, Manuel S Rudolph, M Cerezo, Zoë Holmes, and Hsin-Yuan Huang. Classically estimating observables of noiseless quantum circuits. *arXiv preprint arXiv:2409.01706*, 2024.
- [87] Guillermo González-García, J Ignacio Cirac, and Rahul Trivedi. Pauli path simulations of noisy quantum circuits beyond average case. *Quantum*, 9:1730, 2025.
- [88] Tomislav Begušić, Kasra Hejazi, and Garnet Kin Chan. Simulating quantum circuit expectation values by clifford perturbation theory. *The Journal of Chemical Physics*, 162(15), 2025.
- [89] Uwe Dorner, Rafal Demkowicz-Dobrzanski, Brian J Smith, Jeff S Lundeen, Wojciech Wasilewski, Konrad Banaszek, and Ian A Walmsley. Optimal quantum phase estimation. *Physical review letters*, 102(4):040403, 2009.

- 594 [90] Youle Wang, Lei Zhang, Zhan Yu, and Xin Wang. Quantum phase processing and its applications
595 in estimating phase and entropies. *Physical Review A*, 108(6):062413, 2023.
- 596 [91] Hsin-Yuan Huang, Richard Kueng, Giacomo Torlai, Victor V Albert, and John Preskill. Provably
597 efficient machine learning for quantum many-body problems. *arXiv preprint arXiv:2106.12627*,
598 2021.
- 599 [92] Xun Gao and Lu-Ming Duan. Efficient representation of quantum many-body states with deep
600 neural networks. *Nature Communications*, 8(1):662, 2017.
- 601 [93] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez,
602 Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information*
603 *processing systems*, 30, 2017.
- 604 [94] JC Ferreira and VA2501172 Menegatto. Eigenvalues of integral operators defined by smooth
605 positive definite kernels. *Integral Equations and Operator Theory*, 64(1):61–81, 2009.
- 606 [95] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: convergence and
607 generalization in neural networks. In *Proceedings of the 32nd International Conference on*
608 *Neural Information Processing Systems*, pages 8580–8589, 2018.
- 609 [96] Giacomo Torlai and Matthew Fishman. PastaQ: A package for simulation, tomography and
610 analysis of quantum computers, 2020. URL <https://github.com/GTorlai/PastaQ.jl/>.
- 611 [97] Matthew Fishman, Steven R. White, and E. Miles Stoudenmire. The ITensor Software Library
612 for Tensor Network Calculations. *SciPost Phys. Codebases*, page 4, 2022. doi: 10.21468/
613 SciPostPhysCodeb.4. URL <https://scipost.org/10.21468/SciPostPhysCodeb.4>.
- 614 [98] A Paszke. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint*
615 *arXiv:1912.01703*, 2019.
- 616 [99] Utkarsh Azad and Stepan Fomichev. PennyLane quantum chemistry datasets. [https://](https://pennylane.ai/datasets/h4-molecule)
617 pennylane.ai/datasets/h4-molecule, 2023.